

Adaptive Rejection Sampling

Description

`ars` samples observations from a log-concave distribution using the adaptive rejection sampling (ARS) algorithm.

Usage

```
ars(f, n=100, left_bound = -Inf, right_bound = Inf, x_init, ...)
```

Arguments

<code>f</code>	The function name.
<code>n</code>	The number of observations generated.
<code>Right (Left) bound</code>	If the function is bounded, enter the right (left) bound. Default is an unbounded distribution.
<code>x_init</code>	The initial starting values (see details).
<code>...</code>	Extra parameters to be passed to <code>f</code> .

Details

ARS requires the function `f` be log-concave. An error message will be returned If the function is not log-concave.

The `x_init` must be a vector with length greater than 1 and should be sorted in increasing order, if not, the program will sort the vector. These starting points should not be too close together. If there is no left bound, the derivative at the first point in `x_init` must be positive. Likewise, if there is no right bound, the derivative at the last point in `x_init` must be negative. These are the starting points used by the ARS algorithm.

The output vector of random samples is length `n`. By default, 100 observations are generated.

The function is unbounded by default unless the left/right bounds are specified.

The parameter vector can be included for parameterized functions. The default either uses the parameters already specified by the user or the function's default parameters.

Value

The output is a vector of sampled values from the density f . The length of the vector is determined by n .

References

Gilks, W. R. and Wild, P. (1992), "Adaptive Rejection Sampling for Gibbs Sampling," *Applied Statistics*, Vol. 41, Issue 2, pp. 337-348.

Examples

```
ars(dnorm, 100, x_init=c(-1,1))

ars(dnorm, 100, x_init=c(1,3), mean = 2, sd = 5)

#To check if the function can catch initial points that are too close together
ars_sample <- ars(dnorm, 10000, x_init=c(1,2))
ars_sample <- ars(dnorm, 10000, left_bound=0.1, x_init=c(1,2))
```